

**In the Claims:**

Please amend the claims as follows.

1. (currently amended) A pipeline accelerator, comprising:  
a memory; and  
a hardwired-pipeline circuit coupled to the memory, including at least one processing pipeline, and operable, without executing a program instruction[[to]]:
  - to receive a message that includes data and that includes a header having information indicating a destination of the data by receiving the data and the information on at least one common bus line,
  - to extract the data from the message,
  - to load the extracted data into the memory,
  - to retrieve the extracted data from the memory,
  - to process the retrieved data with a pipeline corresponding to the destination  
~~without executing a program instruction,~~ and
  - to provide the processed data to an external source.
2. (original) The pipeline accelerator of claim 1 wherein:  
the memory is disposed on a first integrated circuit; and  
the pipeline circuit is disposed on a second integrated circuit.
3. (original) The pipeline accelerator of claim 1 wherein the pipeline circuit is disposed on a field-programmable gate array.
4. (original) The pipeline accelerator of claim 1 wherein the pipeline circuit is operable to provide the processed data to the external source by:  
loading the processed data into the memory;  
retrieving the processed data from the memory; and  
providing the retrieved processed data to the external source.

5. (original) The pipeline accelerator of claim 1 wherein:  
the external source comprises a processor; and  
the pipeline circuit is operable to receive the data from the processor.

6. (currently amended) A computing machine, comprising:  
a processor operable to broadcast a message that includes data and that includes a header having information indicating a destination of the data; and  
a pipeline accelerator coupled to the processor and comprising:  
a memory, and  
a hardwired-pipeline circuit coupled to the memory, including at least one processing pipeline, and operable, without executing a program instruction[[ to]]:  
to receive the message from the processor by receiving the data and the information via at least one same bus line,  
to extract the data from the message,  
to load the extracted data into the memory,  
to retrieve the extracted data from the memory,  
to process the retrieved data with a pipeline corresponding to the destination ~~without executing a program instruction~~, and  
to provide the processed data to the processor.

7. (currently amended) A pipeline accelerator, comprising:  
a memory; and  
a hardwired-pipeline circuit coupled to the memory and operable, without executing a program instruction[[ to]]:  
to receive data,  
to process the received data ~~without executing a program instruction~~,  
to load the processed data into the memory,  
to retrieve the processed data from the memory,  
to generate a message header that includes first information indicating a destination of the processed data;

to generate a message that includes the processed data and the header; and  
to provide the message to an external source.

8. (currently amended) A computing machine, comprising:  
a processor operable to run at least one software application; and  
a pipeline accelerator coupled to the processor and comprising:

a memory, and

a hardwired-pipeline circuit coupled to the memory and operable, without  
executing a program instruction: [[ to, ]]

to receive data from the processor,

to process the received data without executing a program instruction,

to load the processed data into the memory,

to generate a message header that includes, for the processed data,  
information that indicates a destination software application running on the  
processor,

to retrieve the processed data from the memory,

to generate a message that includes the retrieved processed data and  
the message header, and

to provide the message to the processor.

9. (currently amended) A pipeline accelerator, comprising:  
first and second memories; and  
a hardwired-pipeline circuit coupled to the first and second memories and  
comprising:

an input-data handler operable without executing a program instruction to  
receive from an external source a first message that includes raw data and that  
includes a first header having information indicating a destination of the raw data, to  
extract the raw data from the message, and to load the raw data into the first  
memory,

at least one hardwired pipeline operable without executing a program  
instruction to process data ~~without executing a program instruction,~~

a pipeline interface operable without executing a program instruction to retrieve the raw data from the first memory, to provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and to load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable without executing a program instruction to retrieve the processed data from the second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source by providing the processed data and the first information to the external source via at least one same bus line.

10. (original) The pipeline accelerator of claim 9 wherein:  
the first and second memories each include respective first and second ports;  
the input-data handler is operable to load the raw data via the first port of the first memory,  
the pipeline interface is operable to retrieve the raw data via the second port of the first memory and to load the processed data via the first port of the second memory, and  
the output-data handler is operable to retrieve the processed data via the second port of the second memory.

11. (original) The pipeline accelerator of claim 9, further comprising:  
a third memory coupled to the hardwired-pipeline circuit;  
wherein the hardwired pipeline is operable to generate intermediate data while processing the raw data; and  
wherein the pipeline interface is operable to load the intermediate data into the third memory and to retrieve the intermediate data from the third memory.

12. (original) The pipeline accelerator of claim 9 wherein:  
the first and second memories are respectively disposed on first and second integrated circuits; and  
the pipeline circuit is disposed on a field-programmable gate array.

13. (original) The pipeline accelerator of claim 9, further comprising:  
an input-data queue coupled to the input-data handler and the pipeline interface;  
wherein the input-data handler is operable to load into the input-data queue a pointer  
to a location of the raw data within the first memory; and  
wherein the pipeline interface is operable to retrieve the raw data from the location  
using the pointer.

14. (currently amended) A pipeline accelerator, comprising:  
first and second memories;  
a hardwired-pipeline circuit coupled to the first and second memories and  
comprising:  
an input-data handler operable to receive from an external source a first  
message that includes raw data and that includes a first header having information  
indicating a destination of the raw data, to extract the raw data from the message,  
and to load the raw data into the first memory,  
at least one hardwired pipeline operable to process data without executing a  
program instruction,  
a pipeline interface operable to retrieve the raw data from the first memory,  
provide the retrieved raw data to a hardwired pipeline corresponding to the  
destination, and load processed data from the hardwired pipeline into the second  
memory,  
an output-data handler operable to retrieve the processed data from the  
second memory, to generate a second header having first information indicating a  
destination of the processed data, to generate a second message that includes the  
processed data and the second header, and to provide the second message to the  
external source by providing the processed data and the first information to the  
external source via at least one same bus line  
The pipeline accelerator of claim 9,  
further comprising:, and  
an output-data queue coupled to the output-data handler and the pipeline  
interface;

wherein the pipeline interface is operable to load into the output-data queue a pointer to a location of the processed data within the second memory; and

wherein the output-data handler is operable to retrieve the processed data from the location using the pointer.

15. (original) The pipeline accelerator of claim 9, further comprising:

wherein each of the input-data handler, hardwired pipeline, pipeline interface, and output-data handler has a respective operating configuration; and

a configuration manager coupled to and operable to set the operating configurations of the input-data handler, hardwired pipeline, pipeline interface, and output-data handler.

16. (original) The pipeline accelerator of claim 9, further comprising:

wherein each of the input-data handler, hardwired pipeline, pipeline interface, and output-data handler has a respective operating status; and

an exception manager coupled to and operable to identify an exception in the input-data handler, hardwired pipeline, pipeline interface, or output-data handler in response to the operating statuses.

17. – 40. Cancelled.

41. (currently amended) A method, comprising:

receiving a message that includes data and that includes a header having information indicating a destination of the data and having information indicating a size of the message;

extracting the data from the message without executing a program instruction;

loading the extracted data into a memory without executing a program instruction;

retrieving the extracted data from the memory without executing a program instruction;

processing the retrieved data with a hardwired-pipeline circuit that corresponds to the destination of the data without executing a program instruction; and

providing the processed data to an external source without executing a program

instruction.

42. (original) The method of claim 41 wherein providing the processed data comprises:

loading the processed data into the memory;  
retrieving the processed data from the memory; and  
providing the retrieved processed data to the external source.

43. (currently amended) A method, comprising:

processing data with a hardwired-pipeline circuit without executing a program instruction;

loading the processed data into a memory without executing a program instruction;

retrieving the processed data from the memory without executing a program instruction;

generating a header having first information indicating a destination of the processed data without executing a program instruction;

forming a message from the header and the processed data without executing a program instruction; and

providing the message to an external source via a single bus without executing a program instruction.

44. (currently amended) A method, comprising:

receiving from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data;

extracting the raw data from the message without executing a program instruction;

loading the extracted raw data into a first memory without executing a program instruction;

retrieving the extracted raw data from the first memory without executing a program instruction;

processing the retrieved data with a hardwired pipeline without executing a program instruction;

loading the processed data from the hardwired pipeline into a second memory without executing a program instruction;

generating a second header having information indicating a destination of the processed data without executing a program instruction;

retrieving the processed data from the second memory without executing a program instruction;

generating a second message that includes the processed data and the second header without executing a program instruction; and

providing the second message to the external source by providing the processed data and the information to the external source via at least one same bus line without executing a program instruction.

45. (original) The method of claim 44 wherein:

loading the raw data comprises loading the raw data via a first port of the first memory;

retrieving the raw data comprises retrieving the raw data via a second port of the first memory;

loading the processed data comprises loading the processed data via a first port of the second memory; and

providing the processed data comprises retrieving the processed data via a second port of the second memory.

46. (original) The method of claim 44, further comprising:

generating intermediate data with the hardwired pipeline in response to processing the raw data;

loading the intermediate data into a third memory; and

providing the intermediate data from the third memory back to the hardwired pipeline.

47. (original) The method of claim 44, further comprising:

loading into an input-message queue a pointer to a location of the raw data within

the first memory; and

wherein retrieving the raw data comprises retrieving the raw data from the location using the pointer.

48. (currently amended) A method, comprising:  
receiving from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data;  
extracting the raw data from the message;  
loading the extracted raw data into a first memory;  
retrieving the extracted raw data from the first memory;  
processing the retrieved data with a hardwired pipeline without executing a program instruction;  
loading the processed data from the hardwired pipeline into a second memory;  
generating a second header having information indicating a destination of the processed data;  
retrieving the processed data from the second memory;  
generating a second message that includes the processed data and the second header;  
providing the second message to the external source by providing the processed data and the information to the external source via at least one same bus line  
~~The method of claim 44, further comprising:~~

loading into an output-message queue a pointer to a location of the processed data within the second memory; and

wherein retrieving the processed data comprises retrieving the processed data from the location using the pointer.

49. (original) The method of claim 44, further comprising setting parameters for loading and retrieving the raw data, processing the retrieved data, and loading and providing the processed data.

50. (original) The method of claim 44, further comprising determining whether an

error occurs during the loading and retrieving of the raw data, the processing of the retrieved data, and the loading and providing of the processed data.

51. – 65. Cancelled.

66. (currently amended) A pipeline accelerator, comprising:

a memory; and

a hardwired-pipeline circuit coupled to the memory, including at least one processing pipeline, and operable:

to receive a message that includes data and that includes a header having information indicating a destination of the data by receiving the data and the information on at least one common bus line,

to extract the data from the message,

to load the extracted data into the memory,

to retrieve the extracted data from the memory,

to process the retrieved data with a pipeline corresponding to the destination without executing a program instruction,

to provide the processed data to an external source~~The pipeline accelerator of claim~~

~~1 wherein the hardwired-pipeline circuit is further operable to:~~

to extract from the header the information indicating the destination of the data;

to generate from the extracted information an identifier that identifies the pipeline corresponding to the destination;

to store the identifier in association with the data; and

to provide the retrieved data to the pipeline in response to the stored identifier.

67. (currently amended) A pipeline accelerator, comprising:

a memory; and

a hardwired-pipeline circuit coupled to the memory, including at least one processing pipeline, and operable:

to receive a message that includes data and that includes a header having

information indicating a destination of the data by receiving the data and the information on at least one common bus line,

to extract the data from the message,

to load the extracted data into the memory,

to retrieve the extracted data from the memory,

to process the retrieved data with a pipeline corresponding to the destination without executing a program instruction,

to provide the processed data to an external source. ~~The pipeline accelerator of claim 1 wherein the hardwired pipeline circuit is further operable to:~~

to extract from the header the information indicating the destination of the data<sub>1</sub>[[:]]

to generate from the extracted information an identifier that identifies the pipeline corresponding to the destination<sub>1</sub>[[:]]

to store a pointer to the extracted data<sub>1</sub>[[:]]

to store the identifier in association with the pointer<sub>1</sub>[[:]] and

to provide the retrieved data to the pipeline in response to the stored pointer and identifier.

68. (currently amended) A pipeline accelerator, comprising:

a memory; and

a hardwired-pipeline circuit coupled to the memory, including at least one processing pipeline, and operable without executing a program instruction[[ to]]:

to receive a message that includes data and that includes a header having information indicating a destination of the data,

to extract the data from the message,

to load the extracted data into the memory,

to retrieve the extracted data from the memory,

to process the retrieved data with a pipeline corresponding to the destination without executing a program instruction,

to provide the processed data to an external source,

to extract from the header the information indicating the destination of the

data,

to store a pointer to the extracted data in a location associated with the pipeline corresponding to the destination, and

to provide the retrieved data to the pipeline in response to the stored pointer.

69. (previously presented) The pipeline accelerator of claim 7 wherein the hardwired-pipeline circuit is further operable to:

store in association with the processed data second information indicating the destination of the processed data;

generate the message header in response to the second information.

70. (previously presented) The pipeline accelerator of claim 69 wherein the second information equals the first information.

71. (currently amended) A pipeline accelerator, comprising:

a memory; and

a hardwired-pipeline circuit coupled to the memory and operable without executing a program instruction[[ to]]:

to receive data,

to process the received data ~~without executing a program instruction~~,

to load the processed data into the memory,

to retrieve the processed data from the memory,

to generate a message header that includes first information indicating a destination of the processed data,

to generate a message that includes the processed data and the header,

to provide the message to an external source,

to store a pointer to the processed data,

to store in association with the pointer second information indicating the destination of the processed data,

to retrieve the processed data in response to the pointer, and

to generate the message header in response to the second information.

72. (currently amended) A pipeline accelerator, comprising:

a memory; and

a hardwired-pipeline circuit coupled to the memory and operable without executing a program instruction[[ to]]:

to receive data,

to process the received data without executing a program instruction,

to load the processed data into the memory,

to retrieve the processed data from the memory,

to generate a message header that includes first information indicating a destination of the processed data,

to generate a message that includes the processed data and the header,

to provide the message to an external source,

to store a pointer to the processed data in a location associated with the destination of the processed data,

to retrieve the processed data in response to the pointer, and

to generate the message header in response to the location.

73. (currently amended) A pipeline accelerator, comprising:

first and second memories;

a hardwired-pipeline circuit coupled to the first and second memories and comprising:

an input-data handler operable to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable to process data without executing a program instruction,

a pipeline interface operable to retrieve the raw data from the first memory, provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and load processed data from the hardwired pipeline into the second

memory, and

an output-data handler operable to retrieve the processed data from the second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source by providing the processed data and the first information to the external source via at least one same bus line;~~The pipeline accelerator of claim 9~~  
wherein[[[:]] the input-data handler is further operable[[ to]]:

to extract from the header the information indicating the destination of the data,

to generate from the extracted information an identifier that identifies the pipeline corresponding to the destination, and

to store the identifier in association with the data; and

wherein the pipeline interface is further operable to provide the retrieved data to the pipeline in response to the stored identifier.

74. (currently amended) A pipeline accelerator, comprising:  
first and second memories; and  
a hardwired-pipeline circuit coupled to the first and second memories and  
comprising:

an input-data handler operable to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable to process data without executing a program instruction,

a pipeline interface operable to retrieve the raw data from the first memory, provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable to retrieve the processed data from the

second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source,

wherein the input-data handler is further operable: [[ to,]]

to extract from the header the information indicating the destination of the data,

to generate from the extracted information an identifier that identifies the pipeline corresponding to the destination,

to store a pointer to the extracted data, and

to store the identifier in association with the pointer; and

wherein the pipeline interface is further operable to provide the retrieved data to the pipeline in response to the stored pointer and identifier.

75. (currently amended) A pipeline accelerator, comprising:  
first and second memories; and  
a hardwired-pipeline circuit coupled to the first and second memories and comprising:

an input-data handler operable, without executing a program instruction, to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable, without executing a program instruction, to process data ~~without executing a program instruction~~,

a pipeline interface operable, without executing a program instruction, to retrieve the raw data from the first memory, to provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and to load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable, without executing a program instruction, to retrieve the processed data from the second memory, to generate a second header

having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source;

wherein the input-data handler is further operable, without executing a program

instruction: [[ to, ]]

to extract from the header the information indicating the destination of the data, and

to store a pointer to the extracted data in a location associated with the pipeline corresponding to the destination; and

wherein the pipeline interface is further operable, without executing a program

instruction, to provide the retrieved data to the pipeline in response to the stored pointer.

76. (currently amended) A pipeline accelerator, comprising:

first and second memories; and

a hardwired-pipeline circuit coupled to the first and second memories and

comprising:

an input-data handler operable, without executing a program instruction, to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable to process data without executing a program instruction,

a pipeline interface operable, without executing a program instruction, to retrieve the raw data from the first memory, to provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and to load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable, without executing a program instruction, to retrieve the processed data from the second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to

provide the second message to the external source;

wherein the pipeline interface is further operable, without executing a program instruction, to store in association with the processed data second information indicating the destination of the processed data; and

wherein the output-data handler is further operable, without executing a program instruction, to generate the first information from the second information.

77. (previously presented) The pipeline accelerator of claim 76 wherein the second information equals the first information.

78. (currently amended) A pipeline accelerator, comprising:

first and second memories; and

a hardwired-pipeline circuit coupled to the first and second memories and comprising:

an input-data handler operable, without executing a program instruction, to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable to process data without executing a program instruction,

a pipeline interface operable, without executing a program instruction, to retrieve the raw data from the first memory, to provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and to load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable, without executing a program instruction, to retrieve the processed data from the second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source;

wherein the pipeline interface is further operable, without executing a program

instruction[[ to]]:

to store a pointer to the processed data, and  
to store in association with the pointer second information indicating the destination of the processed data; and  
wherein the output-data handler is further operable, without executing a program

instruction[[ to]]:

to retrieve the processed data in response to the pointer, and  
to generate the first information from the second information.

79. (currently amended) A pipeline accelerator, comprising:  
first and second memories; and  
a hardwired-pipeline circuit coupled to the first and second memories and comprising:

an input-data handler operable, without executing a program instruction, to receive from an external source a first message that includes raw data and that includes a first header having information indicating a destination of the raw data, to extract the raw data from the message, and to load the raw data into the first memory,

at least one hardwired pipeline operable to process data without executing a program instruction,

a pipeline interface operable, without executing a program instruction, to retrieve the raw data from the first memory, to provide the retrieved raw data to a hardwired pipeline corresponding to the destination, and to load processed data from the hardwired pipeline into the second memory, and

an output-data handler operable, without executing a program instruction, to retrieve the processed data from the second memory, to generate a second header having first information indicating a destination of the processed data, to generate a second message that includes the processed data and the second header, and to provide the second message to the external source;

wherein the pipeline interface is operable, without executing a program instruction, to store a pointer to the processed data in a location associated with the destination of the

processed data; and

wherein the output-data handler is further operable, without executing a program instruction: [[ to]]

to retrieve the processed data in response to the pointer, and

to generate the first information in response to the location.

80. (currently amended) A method, comprising:

receiving a message that includes data and that includes a header having information indicating a destination of the data and having information indicating a size of the message;

extracting the data from the message;

loading the extracted data into a memory;

retrieving the extracted data from the memory;

processing the retrieved data with a hardwired-pipeline circuit that corresponds to the destination of the data without executing a program instruction;

providing the processed data to an external source; ~~The method of claim 41, further comprising:~~

extracting from the header the information indicating the destination of the data;

generating from the extracted information an identifier that identifies the hardwired-pipeline circuit corresponding to the destination;

storing the identifier in association with the data; and

providing the retrieved data to the hardwired-pipeline circuit in response to the stored identifier.

81. (previously presented) A method, comprising:

receiving a message that includes data and that includes a header having information indicating a destination of the data;

extracting the data from the message;

loading the extracted data into a memory;

retrieving the extracted data from the memory;

processing the retrieved data with a hardwired-pipeline circuit that corresponds to

the destination of the data without executing a program instruction;  
providing the processed data to an external source;  
extracting from the header the information indicating the destination of the data;  
generating from the extracted information an identifier that identifies the  
hardwired-pipeline circuit corresponding to the destination;  
storing a pointer to the extracted data;  
storing the identifier in association with the pointer; and  
providing the retrieved data to the hardwired-pipeline circuit in response to the stored  
pointer and identifier.

82. (currently amended) A method, comprising:  
receiving a message that includes data and that includes a header having  
information indicating a destination of the data;  
extracting the data from the message without executing a program instruction;  
loading the extracted data into a memory without executing a program instruction;  
retrieving the extracted data from the memory without executing a program  
instruction;  
processing the retrieved data with a hardwired-pipeline circuit that corresponds to  
the destination of the data and without executing a program instruction;  
providing the processed data to an external source without executing a program  
instruction;  
extracting from the header the information indicating the destination of the data  
without executing a program instruction;  
storing a pointer to the extracted data in a location associated with the  
hardwired-pipeline circuit corresponding to the destination without executing a program  
instruction; and  
providing the retrieved data to the hardwired-pipeline circuit in response to the stored  
pointer without executing a program instruction.

83. (previously presented) The method of claim 43, further comprising:  
storing in association with the processed data second information indicating the

destination of the processed data; and

wherein generating the header comprises generating the header in response to the second information.

84. (currently amended) A method, comprising:

processing data with a hardwired-pipeline circuit without executing a program instruction;

loading the processed data into a memory without executing a program instruction;

retrieving the processed data from the memory without executing a program instruction;

generating a header having first information indicating a destination of the processed data without executing a program instruction;

forming a message from the header and the processed data without executing a program instruction;

providing the message to an external source without executing a program instruction;

storing a pointer to the processed data without executing a program instruction;

storing in association with the pointer second information indicating the destination of the processed data without executing a program instruction;

retrieving the processed data in response to the pointer without executing a program instruction; and

wherein generating the header comprises generating the header in response to the second information.

85. (currently amended) A method, comprising:

processing data with a hardwired-pipeline circuit without executing a program instruction;

loading the processed data into a memory without executing a program instruction;

retrieving the processed data from the memory without executing a program instruction;

generating a header having first information indicating a destination of the processed

data without executing a program instruction;

forming a message from the header and the processed data without executing a program instruction;

providing the message to an external source without executing a program instruction;

storing a pointer to the processed data in a location associated with the destination of the processed data without executing a program instruction;

retrieving the processed data in response to the pointer without executing a program instruction; and

wherein generating the header comprises generating the header in response to the location.